

ACCESS
A PROGRAM FOR THE CATALOG AND ACCESS OF INFORMATION

BY
J. GERRY PURDY

STAN-CS-71-210
MARCH, 1971

COMPUTER SCIENCE DEPARTMENT

School of Humanities and Sciences

STANFORD UNIVERSITY

•

ACCESS

A Program for the Catalog and Access of Information

by

J. Gerry Purdy

The preparation of this report has been supported by IBM Corporation.

1. Abstract

ACCESS is a program for the catalog and access of information. The program is primarily designed for and intended to handle a personal library, although larger applications are possible. ACCESS produces a listing of all entries by locator code (so one knows where to find the entry in his library), a listing of entry titles by user-specified category codes, and a keyword-in-context KWIC listing (each keyword specified by the user). ACCESS is presently programmed in FORTRAN and operates on any IBM System/360 under OS (it uses the IBM SORT/MERGE package). It is anticipated a machine language version (soon to be implemented) will greatly decrease the running time of the program.

2. Introduction

Numerous processes have been developed for cataloging and accessing information. The card catalog in any library is a typical example. When these become large and unwieldy, elaborate automated systems have been devised. For one's private library in his office, such automated systems are uneconomical and impractical. As a result, one is usually left to keep his own card catalog, e.g. on 3 x 5 cards in a file.

When one desires to access the information in a library (personal or public) concerning a particular subject, he has two problems:

- (1) finding all entries in the library pertaining to that subject and
- (2) actually locating the entries of interest. The latter process is the easier to solve. One simply creates a sequential ordering scheme for the file: the next entry to be inserted takes the next sequential assignment code, e.g. the set of integers or some modified code like Uxxxx, where "U" might stand for "unpublished reports" and "xxxx" for "the sequential entry in that file". The choice is arbitrary and can be varied according to the needs of the library.

The other problem -- that of locating all entries in a library pertinent to a particular subject -- quickly becomes quite complex to handle as the number of library entries increases. To create a subject-oriented index means one has to compose a "card" about the entry for each subject it is concerned with.

It becomes evident that there is a need to provide a simple, relatively economical system to provide cataloging and indexing for one's private library. The ACCESS program is an attempt to satisfy that need.

ACCESS takes almost free-form input card images and generates three kinds of reports:

1. ACCESSION LISTING: a listing of each entry in the library by locator (accession) code. This listing provides the master reference showing all input card images for each entry.
2. CATEGORY LISTING: for each category, a listing of each title card belonging to that category.

3. KEYWORD LISTING: a keyword-in-context (KWIC) listing of all keywords giving forward and backward text as additional context.

Examples of typical ACCESS output appear at the end of this report.

3. Input Specification

The ACCESS program may be run in one of two modes of operation: category mode and no-category mode. The category mode gives the full output capability of the program and requires the user to specify his categories. The no-category mode does not list title cards by categories and the codes are not input (see examples in Figures 1 and 2).

The inputs to ACCESS can be given in the following BNF-like notation:

```
*LIB <options>
*CATEGORY NAMES
    <category code> <category images>
*END CATEGORY NAMES
*ACCESSION ENTRIES
    <sets of accession entries>
*END ACCESSION ENTRIES
```

} Not input if
ACCESS run in
no-category mode.

where

<options> contains the options of the program as follows:

NOCAT	means that the run is in the no-category mode of operation and category names and codes (above) are not input
NOLIST	means the input cards will not be listed on the output stream.
(blank)	means category mode and list inputs.

<category code> is the desired category code, specified as any single character available on the 029 keypunch (one of 64 characters) punched in card column 1.

<category images> are the desired category names, up to 79 characters long and punched in card columns 2 through 80.

<sets of accession entries> are the accession entries, each one composed of a title card and one or more continuation cards as necessary; these are described in more detail below.

Keywords

All keywords are specified by the user by placing an @ (8 - 4 punch) immediately before each desired keyword. During listings, all @ are replaced by blanks so that extra blanks between words are not required by the user. Every occurrence of an @ will cause a printed line to be generated in the keyword listing.

Accession Entries

Each accession entry is a description for an actual entry in the user's library, and each entry is composed of one or more 80 character fields. An accession entry is composed of a title card and, if necessary, continuation cards. These are defined as follows:

A. Title Card (First Card of Accession Entry)

1. The Locator Code -- This is the user supplied code which serves as a locator for the entry in the library. It may be up to eight (8) non-blank characters beginning in c.c. 1 (the first blank ends the code). One usually would place a locator character here followed by a sequence number. Some examples might be

U2001	(unpublished report 2001)
L70.029	(1970 letters file, letter no. 29)
20575	(just plain sequence 20,575)
100/555	(sub-report 555 of entry 100)

The user may employ this field for any code he feels will help him locate the actual library entry.

2. The Title -- This is the title of the accession entry and may begin anywhere after the first blank following the locator code. The title may be, itself, more than one card long. If categories are included, the first keyword should be preceded by a \$ (11-3-8 punch) instead of an @ ; this causes the ACCESS program to sort the category listing title cards by \$ entries.

It is also treated as an @ and placed in the keyword listing. (This first keyword is generally the author's name or the originating agency for the report; experience has shown it is preferable to treat author's names as keywords instead of making a separate author list.)

3. The Category Code(s) -- If categories are included, then the user may specify up to three one-character categories for this entry in c.c. 78-80. These one-character codes refer to the code given in the category name cards described above. If categories are not input, then the user may extend his title into columns 78-80.

B. Continuation Cards

1. The Continuation Code -- Each continuation card of the accession entry must begin with a + (12-6-8 punch) in c.c. 1.
2. The Continuation Entry -- The description for the library entry may be continued for up to 255 cards. Card columns 2-80 may be used, and the entry is free form. All @ cause keyword entry listings. These entries should be arranged so that the multiple card images will appear in a pleasing format when printed in the accession listing.

Sample ACCESS Inputs

With the inputs defined, some examples can be given. In Figure 1 at the end of the report an example of the category mode inputs are shown. The category codes appear in the title card for each entry. Note the different methods of locator code assignment.

In Figure 2, a sample of the ACCESS inputs are given where no categories are specified. All 80 columns of the title card may be utilized and a \$

is not necessary. The ACCESS entries correspond to the first few entries of the Computer Science Department Library's unpublished report series.

Sample ACCESS Output

When the inputs to ACCESS are merged with the necessary 360 JCL (see Appendix I), the program may be executed. A sample execution is given in Figure 3 for the full capability of the program, i.e. category mode of operation. Shown in Figure 3 are the accession listing, category listing of title cards, and the keyword listing. This output was generated using the input shown in Figure 1.

A sample run of the non-category mode is given in Figure 4. Here, the category listing is eliminated while the accession and keyword listings are similar to Figure 3. The output here was generated using the input shown in Figure 2.

4. Internal Processing and Organization

In this section we define the data structures employed by the ACCESS program and outline the overall operation of the program.

A. Data Structures

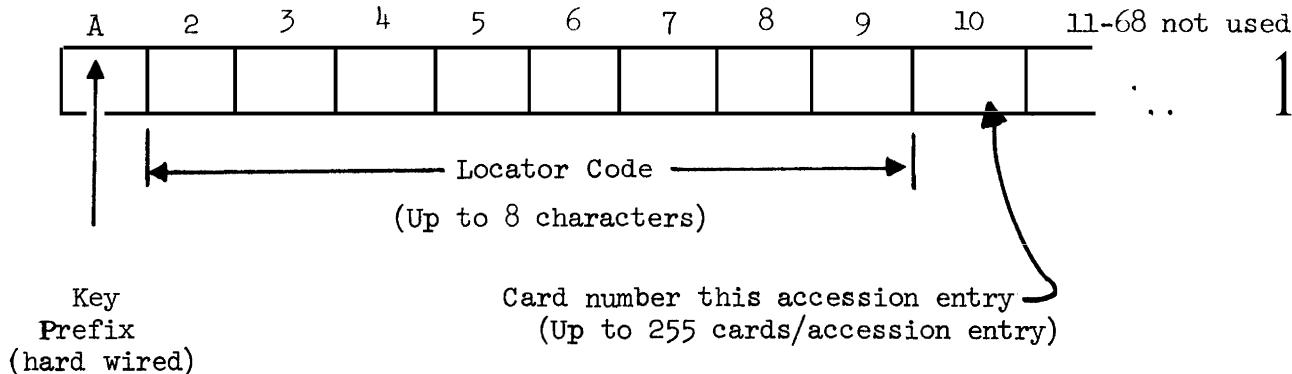
The basic idea behind processing each ACCESS entry is that all print line images for that entry are generated before the next entry is encountered. This is accomplished by the use of the following 200 character data structure, or record:



This structure employs the first 68 characters as a sort key and the remaining 132 characters as the printed line. In the following discussion, it is assumed that the full capability mode is being processed; if it is not, the program simply bypasses the category portion. Only the composition of the sort key is discussed, since the print line structure is easily seen in the output examples (Figures 3 and 4).

1. Accession Listing Sort Key

Of the 68 available characters (8-bit bytes) for the sort key, only the first 10 are utilized. The key takes the following format:



The accession listing is an output of all the ACCESS entries, sorted by locator code. The first character (key prefix) is hard-wired as an "A".

The category entry sort keys all begin with a " B ", and all keyword sort keys begin with a " C ". As a result, all accession records (key prefix equal to " A ") are sorted to the beginning, as desired.

Each accession entry is characterized by its unique locator code. Therefore, the locator code is placed in the sort key immediately after the key prefix (as shown above). Since there may be more than one card image for each entry, the eighth character in the sort key contains the binary equivalent of the card number. The remainder of the characters are not used in the accession listing sort key.

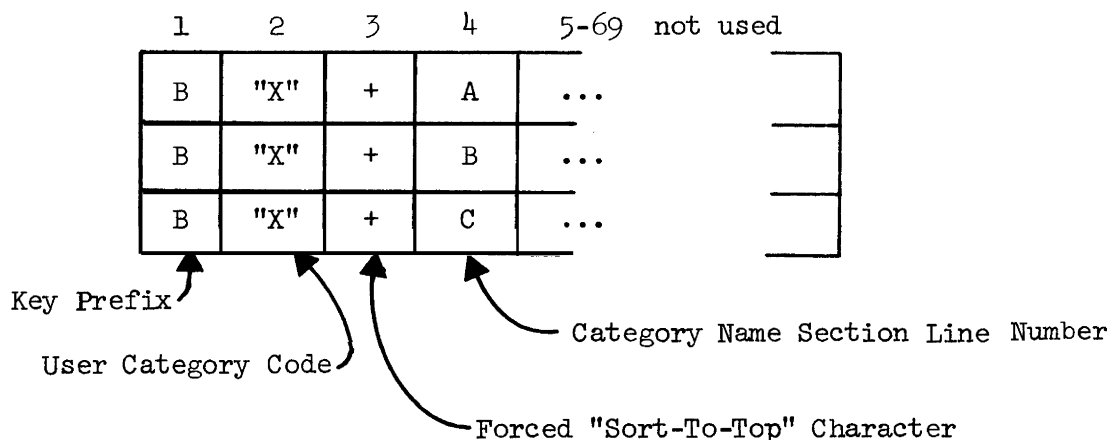
2. Category Listing Sort Key

For the category listing, there are actually two output segments for each category section: the category name segment and the ACCESS entry title card images segment. Each category name is to be set off (with blank lines) and appear at the beginning of each category section. Therefore, the sort key for the category names is different from the sort key for the title card images for that section.

The category name segment of each section is composed of three lines of the form:

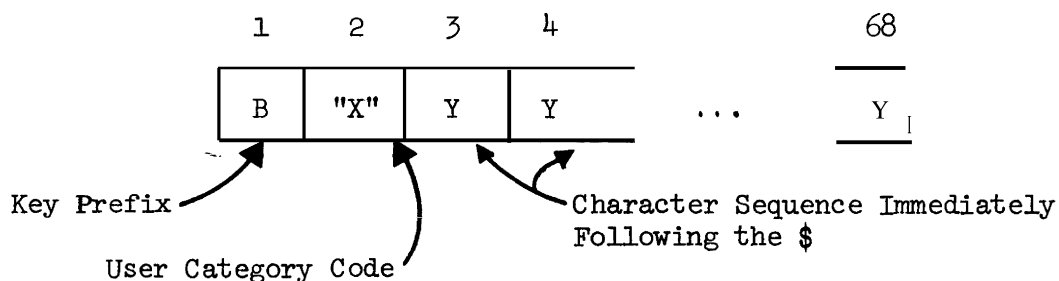
```
(blank)
*****CATEGORY -- X   YYYYYY...Y
(blank)
```

where X is the category code and YYY...Y is the category name. The sort key for these three lines are defined as follows:



Here, the key prefix is a B to indicate the category listing. The "X" is the user category code taken from c.c. 1 of the category names cards. The + (12-6-8 punch) is a character chosen to force the category name segment to the beginning of each category section. If this were not done, then it would be possible for some of the ACCESS entry title cards to sort before the category name. The A , B , C inserted in the fourth character simply serve to uniquely identify the order of the category name segment lines.

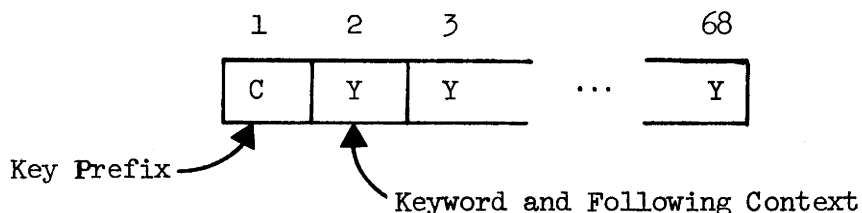
The ACCESS title card image segment contains the first card image in each ACCESS entry that has a category code in columns 78-80 of the first card. Thus, there can be up to three category title images generated for each ACCESS entry. The sort key for each category title image is defined as follows:



As above, the key prefix is a B to indicate a category listing line (or record), and the "X" is the user's category code taken from c.c. 78-80 of the title card in the ACCESS entry. The character sequence YYY...Y is taken from the title card immediately following the \$. If no \$ is present, the character sequence immediately following the locator code is used. The purpose of the YYY...Y sequence is to provide a sort sequence for the title card (usually an author's name, etc.).

3. Keyword Listing Sort Key

The sort key for the keyword listing is quite straightforward and is defined as:



The sort key prefix is a C to indicate the keyword listing. At each occurrence of an @ (a \$ is treated like an @), the character string following the @ is inserted in the sort key until either the sort key is full or the end of the ACCESS entry is reached. Thus, for each keyword (immediately preceded by an @), a keyword listing record is generated.

B. Overall Program Flow

The actual execution of the ACCESS program proceeds in three OS job steps. In the first job step, the input cards are read and a temporary file containing 200 character records (68 character sort key plus 132 character print line) is generated. The second job step (SORT/MERGE) takes the temporary file, sorts it according to the sort keys, and generates another temporary file of sorted 200 character records. The third job step lists the sorted file on the printer.

The processing flow for the first job step (the other two are rather straightforward) begins by reading in the ACCESS *LIB card. The program then decides if there are category names to be read and processed. If there are, they are read. For each category name card, three lines (200 character records) are output. Their sort keys correspond to the category name segment sort key as described above.

After this, the ACCESS entries are processed. The title card for the ACCESS is read, the locator code is stored for later use, and the output record for the accession listing is generated. Next, the category listing entry is composed (unless NOCAT was specified on the *LIB card). Finally, the title card image is placed into the first portion of a 2400 character keyword buffer. A buffer is necessary so that all keyword context may be output for a given keyword; the keyword output records are not generated until either the last card has been sensed or the keyword buffer is full (in which case all keyword records are generated and the buffer is cleared).

All continuation cards are read for the current ACCESS entry and its accession listing output record is made. The continuation card image text is appended to the current end of the keyword buffer. When a new ACCESS entry is read (card image not beginning with a +), the keyword records are composed and output, and the process begins over again.

To demonstrate this process, we can take a sample input deck and show what the keyword portion of the 200 character records look like both before and after the sort. Consider the ACCESS inputs shown in Figure 5. When these have been read in and processed by the first job step, the output records look like those shown in Figure 6. These records are then sorted, and the results are given in Figure 7. One can trace through the above described logic and defined data structures to confirm what is shown.

5. Conclusions

The ACCESS program has been described and sample input and output has been presented. The program is presently coded in FORTRAN IV for the IBM System 360/67 at Stanford. It is anticipated that another version of the program (in machine language) will be developed to decrease its running time.

The ACCESS program provides a free form input which allows one to catalog and subsequently retrieve all information in his private library.

Acknowledgment.

ACCESS is an outgrowth of a very similar, but less versatile, program which had been developed by Prof. D. E. Knuth of Stanford for his personal library.

Appendix I: Operating Instructions, Stanford IBM 360/67

In order to execute the ACCESS program on an IBM 360 under OS, necessary JCL must be specified. A procedure library entry has been made for the complete JCL. As a result, one only needs to generate the following JCL to execute the ACCESS program:

```
//jobname JOB XXXXX
//stepname EXEC ACCESS
//GO.SYSIN DD *
      (ACCESS INPUT CARDS)
/*
```

The JCL as shown above is satisfactory for runs with up to 5,000 ACCESS cards. If more than 5K input cards are anticipated, the user should change his JCL to provide more space for intermediate storage on the disc. See the consultants in the Computation Center for directions concerning how to do this.

Concerning timing estimates, the following sample running times have been recorded on the 360/67:

<u>No. ACCESS Input Cards</u>	<u>No. of Records Generated</u>	<u>Running Time (min.)</u>
52	430	0.76
78	480	0.85
~ 2 K	9.2 K	7.64
-2.5 K	12.5 K	8.50

Appendix II: How to Obtain ACCESS

Stanford Users

For those at Stanford, the ACCESS program has been compiled and placed into a private library and the JCL has been placed in a procedure library (discussed above). The source decks may be obtained from the file &TOOO.ACCESS1 (source for the first job step) and &TOOO.ACCESS2 (source for the third job step). These reside on SYS01.

Caution: If you use the source decks, be sure to supply your private library names.

Non-Stanford Users

For those not at Stanford, the program is available for a nominal charge. Those who desire to obtain a copy of the source deck may write:

User Services
Campus Facility
Stanford Computation Center
Stanford University
Stanford, California 94305

```

1. *LIB NCIST
2. *CATEGORY NAMES
3. A A P T I F I C I A L I N T E L L I G E N C E
4. C C M P I L E R S
5. F F U C A T I O N
6. H H A R D W A R E . . . P E R I P H E R A L S
7. J J E P I G I H C . . . . . T E S T I N G
8. L C M P U T E R L A N G U A G E S
9. N N U M E R I C A L A N A L Y S I S
10. *END CATEGORY NAMES
11. *ACCESSION ENTRIES
12. M0001 $SLAGLE, J.R. AND $BORSKY, P. "EXPER. WITH A MULTI-PURPOSE A
13. + $THEOREM PROVING $HEURISTIC PROGRAM", JOUR. ACM, V O L 15, N O . 1,
14. + PAGES 65-69, 1968. DISCUSSES $MULTIPLE, A DYNAMIC $ORDERING, $BREADTH-FIRST
15. + SEARCHING ALGORITHM WHICH USES $ALPHA-BETA.
16. M0002 $WHITE, C.F., "$TIME-SHARING SERVICES", MODERN DATA, FEB. 1970,
17. + PG. 66. GIVES A TECHNOLOGY SURVEY OF T-S SERVICES.
18. M0003 $GOLDENBERGER, PROF. FRED, "PROGRAM $TESTING, AND $VALIDATING", J
19. + DATAMATION, JULY 1968. GIVES A GOOD SURVEY OF THE GOALS OF PROGRAM TESTING.
20. M0004 $SLAGLE, J.R. AND $DIXON, J.K., "EXPERIMENTS WITH THE $M & $TREE- A
21. + $SEARCHING PROGRAM", ACM, VOL. 13, N O . 3, MARCH 1970, PG. 147-154.
22. + DISCUSSES A VARIATION $MINI-MAX METHOD WHICH AVOIDS $CONUSES F O R $MULTIPLE
23. + SUCCESSOR MAX.'S AND GIVES A PENALTY FOR MULT. SUCCESSOR MIN.'S. THUS
24. + CNETAKES THE FRANCH WITH THE MOST LIKELIHOOD OF SUCCESS IN DEEPER
25. + PENETRATIONS. THE $ALPHA-BETA PROCEDURE ALSO HAD TO BE MODIFIED.
26. W0005 $FENICHEL, R.P., $WITZENBAUM, J., AND $VOCHFLSON, J.C., "A PROGRAM FL
27. + T O $TEACH $PROGRAMMING", ACM, VOL. 13, NO. 3, MARCH 1970, PG. 141-146.
28. + GAVE A SURVEY ABOUT THE $TEACH SYSTEM, USED TO A COURSE IN PROGRAMMING.
29. + THE STUDENTS LEARNED THE $UNCL LANGUAGE.
30. M0006 $SCHWAPZ, R.M., $BURGER, J.F., AND $SIMMONS, R.F., "$ADDEDUCTIVE AI
31. + $QUESTION-ANSWERER FOR $NATURAL LANGUAGE $INFERENCE", ACM, VOL. 13, N O . 3,
32. + MARCH 1970, PG. 167-183. DISCUSSES THE $PROTOSYNTHESIS.
33. M0107 $BRYAN, GLENN L., "$STUDENT-TO-STUDENT $INTERACTION IN COMPUTER $TIME- E
34. + $SHAQFC SYSTEM", C&A, MARCH, 1970, P. 18.
35. M0008 $MORTON, KENT, "THE VALUE OF THE COMPUTER AS $A PUPIL", ARTICLE E
36. + $AN EDUCATION IN MARCH, 1970, C & A, P. 24
37. M0109 $FERGLSON, R.L., "$COMPUTER ASSISTANCE FOR INDIVIDIZED $INSTRUCTION", E
38. + ARTICLE $AN EDUCATION IN MARCH, 1970, C&A, P. 27.
39. M0010 $SCHWETMAN, H.C., "$L S E C F $REMOTE $CONSOLE SYSTEM ON $UNIVERSITY EH
40. + CAMPS", ARTICLE $AN EDUCATION IN MARCH, 1970, C&A, P. 30.
41. R0001 $FENDRAL: $BUCHANAN, B., $SUTHERLAND, G., $FEIGENBAUM, E.A., A
42. + "$HEURISTIC $GENERAL: A PROGRAM FOR GENERATING EXPLANATORY
43. + $HYPOTHESES IN ORGANIC CHEMISTRY", MACHINE INTELLIGENCE 4,
44. + EDINBURGH UNIVERSITY PRESS, 1969.
45. R0002 $GREENBLATT, R.C., ET. AL., "THE GREENBLATT $CHESS PROGRAM", 4
46. + PROC. F J C C, 1967, PG. 801 TO 810.
47. R0003 $MICHIE, D., $FLEMING, J.G., $OLDFIELD, J.V., "A COMPARISON OF A
48. + $HEURISTICS, $INTERACTIVE, AND $UNAIDED METHODS FOR SOLVING $A $HARDEST
49. + $ROUTE PROBLEM", MACHINE INTELLIGENCE 3, EDINBURGH U N I V. PRESS, 1968.
50. R0004 $NEWELL, A., "$HEURISTIC $SEARCH: ILL-STRUCTURED PROBLEMS", PROGRESS 4
51. + $IN OPERATIONS RESEARCH (VOL. 3), WILEY, 1969. DISCUSSES $HEURISTIC METHODS
52. + $IN PROBLEM SOLVING VERY NICELY - GOOD SUMMARY.
53. R0005 $NILSSON, N.J., "$SEARCHING $PROBLEM SOLVING AND $GAME P L A Y I N G $TREES A
54. + $F G R M A T I A L C O S T $SOLUTIONS", PROCEED. IFIP 68 CONGRESS, EDINBURGH, SCOTLAND,
55. + 1968. NOTE - MUCH OF THIS PAPER IS ALSO I N $NILSSON'S BOOK.
56. + DISCUSSES $NILSSON'S ARCH METHOD OF $R $SEARCHING $AND $OR $TREES.
57. U0053 $SESAR, DENNIS, "$HARDWARE-SOFTWARE $EMULATOR FOR THE $IBM $360", H
58. + ENGR. 2250, SPRING 1968, PROF. G. $ESTRIN. USES $AN $XDS $SIGMA $7 $ASEMULATOR.
59. U0054 $KNUTH, D.E. AND $FLOYD, R.W. "NOTES ON AVOIDING $GOTO $STATEMENTS", L
60. + $STANFORD CSD REPORT CS 144, JAN. 1970. CONSIDERS TRANSFORMING A "GOTO"
61. + TYPE PROGRAM TO A $RECURSIVE PROCEDURE TYPE USING $ALGOL'S W H I L E, ETC.
62. U0055 $CATHAN, T.F. "THE THEORY AND CONSTRUCTION OF $COMPILERS", C
63. + $COMPUTER ASSOCIATES, CA-6606-0111, JUNE 2, 1966. HAS $GRAMMARS, $SYNTAX
64. + $ANALYSIS, $LEXICAL ANALYSIS, $DEFINITIONAL FACILITIES ($EXTENDABILITY) AND
65. + $OPTIMIZATION.
66. U0056 $FORSYTHE, G.F. "PITFALLS IN $COMPUTATION, OR WHY $A $MATH $BOOK I $N'T N
67. + $ENOUGH", $STANFORD CSD REPORT CS 147, J A N, 1970. $NUMERICAL ANALYSIS,
68. + $ROUND OFF, $TRUNCATION.
69. U0058 $CATHAN, T.F. "THE THEORY A N D CONSTRUCTION OF $COMPILERS", 1967 C
70. + $REPORT FROM $COMPUTER ASSOCIATES ON $SYNTAX ANALYSIS. N O N O. OR INDEX.
71. *END ACCESSION ENTRIES
72.
73.
74.

```

Figure 1. Sample Category Mode ACCESS Input

```

*LIB NOCAT
*ACCESSION ENTRIES
000001 ABUSO, C.*ANALOG COMPUTATION WITH RANDOM PULSE SEQUENCES.*ILLINOIS U
+COMP.SCI.DEPT.255.*1968.
000002 AMERICAN STANDARDS ASS'N.*WORKING DOCUMENT ACOROL
+SPECIFICATIONS*AMERICAN STANDARDS ASS'N.*1965.
000003 ABRAHAM, P.*LISP-2 LANGUAGE SPECIFICATIONS.*SYSTEM OVERFL OPMENT
+COMP.*1M-3417/200/00.*1967.
000004 APELKANS, M.Y.*ON DIFFERENCE SCHEMES FOR HYPERBOLIC-EQUATIONS WITH
+DISCONTINUOUS INITIAL-VALUES.*JUPPSALA U. COMP.SCI.DEPT.*RPT.5.*1967.
000005 ANSELONE, P.M.*CORE, R.M.*AN EXTENSION OF THE NEWTON-KANTOROVICH
+METHOD FOR SOLVING NONLINEAR-EQUATIONS.*MILWAUKEE U. MATH.RES.CENTER.*MRC
+TECH. SUMMARY RPT. 520.*1965.
000006 ANDERSON, N.S.*SCENARIOS, A.S. IMENSUN, N.F.*PATTERN-RECOGNITION: I. A
+COMPUTER PROGRAM FOR GENERATING SYNTHETIC PATTERNS.*MARYLAND U. COMP.SCI.
+CENTER.*TR-64-8.*1964.
000007 APPLIED LOGIC CORP.*SEMI-ANNUAL REPORT: GT RAI DECEMBER-AUTOMATED
+MATHEMATICAL LOGIC APPLIED LOGIC CORP.*1967.
000008 ARBIB, M.A.*SOME THEORETICAL APPROACHES TO NERVOUS-SYSTEM FUNCTION.*
+7N1UFDIY BEONETICS*3ST. U.F.E.DEPT.*1967.
000009 BALZER, R.M.*STUDIES CONCERNING MINIMAL TIME SOLUTIONS TO
+TRIP LING-SQUAD SYNCHRONIZATION PROBLEM.*CARNEGIE INST. OF TECHNOLOGY.*
+1964.
000010 BAERCKER, P.M.*PLANAR REPRESENTATIONS OF COMPLEX GRAPHS.*M.I.T.
+LINCOLN LAB.*T.N. 1967-1.*1967.
000011 ARBIB, M.A.*GIVEN, Y.*CONTRIBUTIONS TO THE THEORY OF ALGEBRAS
+CONSIDERED AS AUTOMATA.*ST. U.F.E.DEPT.*1967.
000012 BALUEV, A.N.*THREE PAPERS ON CHAPLYGIN-METHODS
+AND NONLINEAR-EQUATIONS.*MARYLAND U. COMP.SCI. CENTER.*TR-67-41.*1967.
000013 ATKINS, D.E.*THE THEORY OF IMPLEMENTATION OF SRT DIVISION.*ILLINOIS
+U. COMP.SCI. DEPT.*RPT. 230.*1967.
000014 ARNOLD, L.*ON THE ASYMPTOTIC DISTRIBUTION OF THE EIGENVALUES OF
+RANDOM MATRICES.*MILWAUKEE U. MATH.RES. CENTER.*MRC TECH. SUMMARY RPT.
+736.*1967.
000015 ARBIB, M.A.*KAHN, R.M.*A DEVELOPMENTAL MODEL
+FOR INFORMATION PROCESSING IN THE CHILD.*ST. U.F.E.DEPT.*1967.
000016 ARBIB, M.A.*KAHN, R.M.*CYBERNETIC APPROACH TO MENTAL DEVELOPMENT.*
+ST. U.F.E.DEPT.*1967.
000017 ARBIB, M.A.*NOTES ON A PARTIAL SURVEY OF CYBERNETICS IN EUROPE AND
+THE U.S.S.R.*P.F.C. RESEARCH ASSOCIATES, INC.*AFOSR-65-1412.*1965.
000018 ARBIB, M.A.*SELF-REPRODUCING AUTOMATA: SOME IMPLICATIONS FOR
+THEORETICAL BIOLOGY.*ST. U.F.E.DEPT.*1967.
000019 ARBIB, M.A.*GIVEN, Y.*ALGEBRAIC AUTOMATA II: THE CATEGORICAL
+FRAMEWORK FOR DYNAMIC ANALYSIS.*ST. U.F.E.DEPT.*1967.
000020 ABAYER, P.*ON ENDOMORPHISMS AND CONGRUENCE OF AUTOMATA.*BOEING
+SCI. RES. LABS.*MATH. NOTE 407.*1967.
000021 ABAYER, R.*THE AUTOMORPHISM GROUP OF A STRONGLY CONNECTED AUTOMATON
+ADITS QUOTIENT AUTOMATA.*ILLINOIS U. COMP.SCI. DEPT.*RPT.190.*1965.
000022 ABAYER, R.*AUTOMORPHISM GROUPS AND QUOTIENTS OF STRONGLY
+CONNECTED AUTOMATA AND MONADIC-ALGEBRAS.*ILLINOIS U. COMP.SCI. DEPT.*RPT.
+204.*1966.
+END ACCESSIONENTRIES

```

Figure 2. Sample No Category Mode ACCESS inputs

ACCESSION CODE	A C C E S S I O N E N T R Y	CATEGORY CODES
MC001	SLAGLE, J.F. AND BURSKY, P. "EXPER. WITH A MULTI-PURPOSE THEORY PROVING HEURISTIC PROGRAM", JEDP. ACM, VOL 15, NO. 1, PAGES 95-99, 1968. DISCUSSES MULTIPLE, A DYNAMIC ORDERING, BREADTH-FIRST SEARCHING ALGORITHM WHICH USES ALPHA-BETA.	A
MC002	WHITE, C.H., "THE SYAPING SERVICES", MODERN DATA, FEB. 1970, PG. 46. GIVES A TECHNOLOGY SURVEY OF T-S SERVICES.	
MC003	QUENBERGER, PROF. FRED, "PROGRAM TESTING AND VALIDATING", DATACTION, JULY 1968. GIVES A GOOD SURVEY OF THE GOALS OF PROGRAM TESTING.	J
MC004	SLAGLE J.F. AND DIXON, J. K., "EXPERIMENTS WITH THE MENTREE-SEARCHING PROGRAM", CACM, VOL. 13, NO. 3, MARCH 1970, PG. 147-154. DISCUSSES A VARIANT MINI-MAX METHOD WHICH AWARDS BONUSES FOR MULTIPLE SUCCESSOR MAX.'S AND GIVE A PENALTY FOR MULT. SUCCESSOR MIN.'S. THUS GIVE TAKES THE BRANCH WITH THE MOST LIKELIHOOD OF SUCCESS IN DEEPER PENETRATIONS. THEIR ALPHA-BETA PROCEDURE ALSO HAD TO BE MODIFIED.	A
MC005	FENICHEL, R.R., WEITZENBAUM, J., AND DYCKELSON, J.C., "A PROGRAM TO TEACH PROGRAMMING", CACM, VOL. 13, NO. 3, MARCH 1970, PG. 141-146. GAVE A SURVEY ABOUT THE TEACH SYSTEM, USED TO A COURSE IF PROGRAMMING. THE STUDENTS LEARNED THE UNCL LANGUAGE.	E L
MC006	SCHWAPCZ, R.W., BURGER, J.F., AND "SIMMONS, R.F.", "4 DEDUCTIVE QUESTION-ANSWER FOR A NATURAL LANGUAGE INFERENCE", CACM, VOL. 13, NO. 3, MARCH 1970, PG. 167-183. DISCUSSES THEIR PROTO SYNTHESIS II.	A L
MC007	RYAN, GLENN L., "STUDENT-TO-STUDENT INTERACTION IN COMPUTER TIME-SHARED SYSTEM", C&A, MARCH 1970, P. 18.	E
MC008	MORTON, KENT, "THE VALUE OF THE COMPUTER AS A PUPIL", ARTICLE ON EDUCATION IN MARCH, 1970, C&A, P. 24	E
MC009	FERGUSON, P.L., "COMPUTER ASSISTANCE FOR INDIVIDIZED INSTRUCTION", ARTICLE ON EDUCATION IN MARCH, 1970, C&A, P. 27.	E
MC010	SCHWETMAN, H.D., "USE OF A REMOTE CONSOLE SYSTEM ON A UNIVERSITY CAMPUS", ARTICLE ON EDUCATION IN MARCH, 1970, C&A, P. 30.	E H
MC001	GENDRAL: RICHMAN, B., SUTHERLAND, G., FEIGENBAUM, E.A., "HEURISTIC DENDRAL: A PROGRAM FOR GENERATING EXPLANATORY HYPOTHESES IN ORGANIC CHEMISTRY", MACHINE INTELLIGENCE 4, EDINBURGH UNIVERSITY PRESS, 1969.	A
MC002	GREENBLATT, P. D., ET. AL, "THE GREENBLATT CHESS PROGRAM", PROC. FUCC, 1967, PG. 801 TO 810.	A
MC003	MICHIE, D., FLEMING, J.G., OLDFIELD, J.V., "A COMPARISON OF HEURISTICS, INTERACTIVE, AND UNAIDED METHODS FOR SOLVING A SHORTEST	A

Figure 3. Sample Category Mode ACCESS Output (Sheet 1 of 5)

ACCESSION CODE	ACCESS ENTRY	CATEGORY CODES
0004	NEWELL, A. "HEURISTIC SEARCH: ILL-STRUCTURED PROBLEMS", PROGRESS IN OPERATIONS RESEARCH (VOL. 3), WILEY, 1969. DISCUSSES HEURISTIC METHODS IN PROBLEM SOLVING VERY NICELY - GOOD SUMMARY.	A
0005	NILSSON, N.-J., "SEARCHING PROBLEM SOLVING AND GAMEPLAYING TREES FOR MINIMAL COST SOLUTIONS", PROCEED. IFIP 69 CONGRESS, EDINBURGH, SCOTLAND, 1969. NOTE - MUCH OF THIS PAPER IS ALSO IN NILSSON'S BOOK. DISCUSSES NILSSON'S APPROX METHOD FOR SEARCHING AND-OF TREES.	A
0053	SESAR, DENNIS, "A HARDWARE-SOFTWARE EMULATOR FOR THE IBM 360", ENG. 2258, SPRING 1969, PROF. G. ESTRIN. USES AN XOSI GHA 7 AS EMULATOR.	F
0054	KNUTH, D.E. AND FLOYD, R.W. "NOTES ON AVOIDING GO TO STATEMENTS", STANFORD CS REPORT CS 149, JAN. 1970. CONSIDERS TRANSFORMING 4 "GOTO" TYPE PROGRAM TO EQUIVALENT PROCEDURE TYPE USING ALGOL'S WHILE, ETC.	L
0055	CHEATHAM, T. E. "THE THEORY AND CONSTRUCTION OF COMPILERS", COMPUTER ASSOCIATES, CC-5605-0111, JUNE 7, 1966. HAS GRAMMARS, SYNTAX ANALYSIS, LEXICAL ANALYSIS, DEFINITIONAL FACILITIES (EXTENSIBILITY) AND OPTIMIZATION.	C
0056	FORSYTHE, G. E. "PITFALLS IN COMPUTATION, OR WHY A MATH BOOK ISN'T ENOUGH", STANFORD CS REPORT CS 147, JAN. 1970. NUMERICAL ANALYSIS, ROUND-OFF, TRUNCATION.	N
0058	CHEATHAM, T. E. "THE THEORY AND CONSTRUCTION OF COMPILERS", 1967 REPORT FROM COMPUTER ASSOCIATES ON SYNTAX ANALYSIS. NO INDEX.	C

Figure 3. Sample Category Mode ACCESS Output (Sheet 2 of 5)

ACCESS
CODE
C A T E G O R Y L I S T I N G

```

***** CATEGORY -- 4 ...   A R T I F I C I A L   I N T E L L I G E N C E
DENDRAL: BUCHANAN, R., SUTHERLAND, G., FEIGENBAUM, E. A.,
GREENBLATT, R. D., ET AL., "THE GREENBLATT CHESS PROGRAM",
WICHTIE, D., FLEMING, J. G., CLDFIELD, J. V., "A COMPARISON OF
NEWELL, A., "HEURISTIC SEARCH: ILL-STRUCTURED PROBLEMS", PROGRESS
NILSSON, N. J., "SEARCHING PROBLEMS SOLVING AND GAME PLAYING TREES
SCHWARCZ, R. M., RUPGER, J. F., AND SIMMONS, R. F., "A DEDUCTIVE
SLAGLE, J. R. AND BURSKEY, P., "EXPER. WITH A MULTI-PURPOSE
SLAGLE, J. R. AND DIXON, J. K., "EXPERIMENTS WITH THE MENTREE-

***** CATEGORY -- C ...   C O M P I L E R S
CHEATHAM, T. F., "THE THEORY AND CONSTRUCTION OF COMPILERS",
CHEATHAM, T. F., "THE THEORY AND CONSTRUCTION OF COMPILERS", 1967

***** CATEGORY -- E ...   E D U C A T I O N
BRYAN, GLENN L., "STUDENT-TO-STUDENT INTERACTION IN COMPUTER TIME-
FENICHEL, R. R., WEIZENBAUM, J., AND YOCHELSON, J. C., "A PROGRAM
FERGUSON, R. L., "COMPUTER ASSISTANCE FOR INDIVIDUALIZED INSTRUCTION",
MORTON, KENT, "THE VALUE OF THE COMPUTER AS A PUPIL", ARTICLE
SCHWETMAN, H. D., "USE OF A REMOTE CONSOLE SYSTEM ON A UNIVERSITY

***** CATEGORY -- H ...   H A R D W A R E   P E R I P H E R A L S
SCHWETMAN, H. D., "USE OF A REMOTE CONSOLE SYSTEM ON A UNIVERSITY
SESAR, DENNIS, "A HARDWARE-SOFTWARE EMULATOR FOR THE IBM 360".

***** CATEGORY -- J ...   D E B U G G I N G   T E S T I N G
GRUENBERGER, PROF. FRED, "PROGRAM TESTING AND VALIDATING",

***** CATEGORY -- L ...   C O M P U T E R   L A N G U A G E S
FENICHEL, R. R., WEIZENBAUM, J., AND YOCHELSON, J. C., "A PROGRAM
KNUTH, D. E. AND FLOYD, R. M., "NOTES ON AVOIDING GO TO STATEMENTS",
SCHWARCZ, R. M., RUPGER, J. F., AND SIMMONS, R. F., "A DEDUCTIVE

***** CATEGORY -- M ...   N U M E R I C A L   A N A L Y S I S
FOR SYTHE, G. E., "PITFALLS IN COMPUTATION, OR WHY A MATH BOOK ISN'T

R0001
R0002
R0003
R0004
R0005
R0006
M0001
M0004
U0005
U0058
M0007
M0005
M0009
M0008
M0010
M0010
M0010
M0003
M0005
U0054
M0006
U0056

```

Figure 3. Sample Category Mode ACCESS Output (Sheet 3 of 5)

D
 E
 C
 A
 Y
 E
 K

00001	00002	00003	00004	00005	00006	00007	00008	00009	00010	00011	00012	00013	00014	00015	00016	00017	00018	00019	00020	00021	00022	00023	00024	00025	00026	00027	00028	00029	00030	00031	00032	00033	00034	00035	00036	00037	00038	00039	00040	00041	00042	00043	00044	00045	00046	00047	00048	00049	00050	00051	00052	00053	00054	00055	00056	00057	00058	00059	00060	00061	00062	00063	00064	00065	00066	00067	00068	00069	00070	00071	00072	00073	00074	00075	00076	00077	00078	00079	00080	00081	00082	00083	00084	00085	00086	00087	00088	00089	00090	00091	00092	00093	00094	00095	00096	00097	00098	00099	00100	00101	00102	00103	00104	00105	00106	00107	00108	00109	00110	00111	00112	00113	00114	00115	00116	00117	00118	00119	00120	00121	00122	00123	00124	00125	00126	00127	00128	00129	00130	00131	00132	00133	00134	00135	00136	00137	00138	00139	00140	00141	00142	00143	00144	00145	00146	00147	00148	00149	00150	00151	00152	00153	00154	00155	00156	00157	00158	00159	00160	00161	00162	00163	00164	00165	00166	00167	00168	00169	00170	00171	00172	00173	00174	00175	00176	00177	00178	00179	00180	00181	00182	00183	00184	00185	00186	00187	00188	00189	00190	00191	00192	00193	00194	00195	00196	00197	00198	00199	00200	00201	00202	00203	00204	00205	00206	00207	00208	00209	00210	00211	00212	00213	00214	00215	00216	00217	00218	00219	00220	00221	00222	00223	00224	00225	00226	00227	00228	00229	00230	00231	00232	00233	00234	00235	00236	00237	00238	00239	00240	00241	00242	00243	00244	00245	00246	00247	00248	00249	00250	00251	00252	00253	00254	00255	00256	00257	00258	00259	00260	00261	00262	00263	00264	00265	00266	00267	00268	00269	00270	00271	00272	00273	00274	00275	00276	00277	00278	00279	00280	00281	00282	00283	00284	00285	00286	00287	00288	00289	00290	00291	00292	00293	00294	00295	00296	00297	00298	00299	00300	00301	00302	00303	00304	00305	00306	00307	00308	00309	00310	00311	00312	00313	00314	00315	00316	00317	00318	00319	00320	00321	00322	00323	00324	00325	00326	00327	00328	00329	00330	00331	00332	00333	00334	00335	00336	00337	00338	00339	00340	00341	00342	00343	00344	00345	00346	00347	00348	00349	00350	00351	00352	00353	00354	00355	00356	00357	00358	00359	00360	00361	00362	00363	00364	00365	00366	00367	00368	00369	00370	00371	0037
-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	------

Figure 3. Sample Category Mode ACCESS Output (Sheet 5 of 5)

ACCESSION CODE	A C C E S S I O N E N T R Y	C A T E G O R Y C O D E S
CC0001	DEUSO, C.*ANALOG COMPUTATION WITH RANDOM PULSE SEQUENCES.* ILLINOIS U. COMP.SCI,DEPT, RPT, 255,*1963.	
CC0002	AMERICAN STANDARDS ASS'N.*WORKING DOCUMENT COBOL SPECIFICATIONS.* AMERICAN STANDARDS ASS'N.* 1965.	
CC0003	ABRAMAMS, P.* LISP-2 LANGUAGE SPECIFICATIONS.* SYSTEM DEVELOPMENT CORP.*TR-3217/200/00.*1967.	
CC0004	APPELKRAS, M.V.*ON DIFFERENCE SCHEMES FOR HYPERBOLIC-EQUATIONS WITH CONTINUOUS INITIAL-VALUES.* UPSALA U. COMP.SCI,DEPT.*RPT.5.*1967.	
CC0005	AMSELME, P.W. MOORE, R.H.*AN EXTENSION OF THE NEWTON-KANTOROVICH METHOD FOR SOLVING NONLINEAR-EQUATIONS.* WISCONSIN U. MATH.RES. CENTER.*MRC TECH. SUMMAR Y RPT. 520.*1965.	
CC0006	ANDERSON, N.S. ROSENFELD, J. SIMENSON, N.F.*PATTERN-RECOGNITION:1. A COMPUTER PROGRAM FOR GENERATING SYNTHETIC-PATTERNS.* MARYLAND U. COMP.SCI. CENTER # TP-64-9.*1964.	A
CC0007	APPLIED LOGIC CORP.*SEMI-ANNUAL REPORT: CTR-AIDED SEMI-AUTOMATED MATHEMATICS.* SAC001* APPLIED LOGIC CORP.* 1967.	
CC0008	APPLEB, M.A.*SOME THEORETICAL APPROACHES TO NERVOUS-SYSTEM FUNCTION.* NEUROCYBERNETICS* CO. U. E.E.DEPT.*1967.	
CC0009	BALZER, R.M.*STUDIES CONCERNING MINIMAL TIME SOLUTIONS IN THE FIRING-SQUAD SYNCHRONIZATION PROBLEM.* CARNEGIE INST. OF TECHNOLOGY.* 1966.	
CC0010	PARKER, R.M.*PLANAR-REPRESENTATIONS OF COMPLEX GRAPHS.* M.I.T. LINCOLN LAB.*TR. 1947-1*1967.	
CC0011	APR 19, M.A. GIVFON, V.*CONTRIBUTIONS TO THE THEORY OF ALGEBRAS CONSIDERED AS AUTOMATA.* ST. U. E.E.DEPT.*1967.	
CC0012	BALCEV, A.A.*THREE PAPERS ON CHAPLYGIN-METHODS AND NONLINEAR-EQUATIONS. * MARYLAND U. COMP.SCI. CENTER.*TP-67-41.*1967.	
CC0013	ATKINS, D.E.*THE THEORY OF IMPLEMENTATION OF SRT DIVISION.* ILLINOIS U. COMP.SCI,DEPT, RPT. 230.*1967.	
CC0014	ASHTED, L.*ON THE ASYMPTOTIC DISTRIBUTION OF THE EIGENVALUES OF RANDOM MATRICES.* WISCONSIN U. MATH.RES. CENTER.*MRC TECH. SUMMARY RPT. 734.*1967.	
CC0015	AKHIE, V.A. KAHN, P.W.*A DEVELOPMENTAL MODEL OF INFORMATION-PROCESSING IN THE CHILD.* ST. U. E.E.DEPT.* 1967.	
CC0016	AKHIE, V.A. KAHN, P.W.*A CYBERNETIC APPROACH TO MENTAL DEVELOPMENT.*	

Figure 4. Sample No Category Mode ACCESS Output (Sheet 1 of 4)

ACCESSION CODE	ACCESSION ENTRY	CATEGORY CODES
000017	ST. U. E.E. DEPT.* 1967. ARRIF, M.A. NOTES ON A PARTIAL SURVEY OF CYBERNETICS IN EUROPE AND THE U.S.S.R.* P.E.C. RESEARCH ASSOCIATES, INC.* AFOSM-65-1412.* 1965.	
000018	ARRIF, M.A. * SELF-REPRODUCING-AUTOMATA; SOME IMPLICATIONS FOR THEORETICAL LOGIC.* ST. U. E.E. DEPT.* 1967.	
000019	ARRIF, M.A. GIVISON, Y. * ALGEBRA AUTOMATA II: THE CATEGORICAL FRAMEWORK FOR DYNAMIC ANALYSIS.* ST. U. E.E. DEPT.* 1967.	
000020	BAYER, P. * ON HOMOMORPHISMS AND CONGRUENCES OF AUTOMATA.* BOEING SCI. RES. LABS.* MATH. NOTE 497.* 1967.	
000021	BAYER, P. * THE AUTOMORPHISM GROUP OF A STRONGLY CONNECTED AUTOMATON AND ITS QUOTIENT AUTOMATA.* ILLINOIS U. COMP. SCI. DEPT.* RPT. 199.* 1966.	
000022	BAYER, P. * AUTOMORPHISM GROUPS AND QUOTIENTS OF STRONGLY CONNECTED AUTOMATA AND MODULAR ALGEBRAS.* ILLINOIS U. COMP. SCI. DEPT.* RPT. 204.* 1966.	

Figure 4. Sample No Category Mode ACCESS Output (Sheet 2 of 4)

KEYWORD LISTING

APPLHAMS, P. *LISP-2 LANGUAGE SPECIFICATIONS.*SYSTEM 000003
 AFUSG, C.*PARALOG COMPUTATION WITH RANDOM PULSE SEQU 000001
 ALGERRA AUTOMATA II: THE CATEGORICAL FRAMEWORK FOR D 000019
 ALGERRAS CONSIDERED AS AUTOMATA. ST. U.E.E.DEPT.* 000011
 AMERICAN STANDARDS ASSN.*WORKING DOCUMENT COROL 000002
 AMERICAN STANDARDS ASSN.*WORKING DOCUMENT COROL 000002
 ANDERSON, N.S. ROSENFELD, A. STIMENSON, N.F.*PATTERN-R 000006
 ANSE TONE, P.M. MOORE, R.H.*AN EXTENSION OF THE NEWTON 000005
 APELKANS, M.Y.*ON DIFFERENCE SCHEMES FOR HYPERPOLIC-E 000004
 APPLIED LOGIC CORP.*1967. P 000007
 APPLIED LOGIC CORP.*SEMI-ANNUAL REPORT: CTR-AIDED SEMI-000007
 ARBIR, M.A.*SELF-REPRODUCING-AUTOMATA: SOME IMPLICATI 000017
 ARBIR, M.A.*NOTES ON A PARTIAL SURVEY OF CYBERNETICS 1~000017
 ARBIR, M.A.*SOME THEORETICAL APPROACHES TO FRYVUS-SYS 000008
 ARBIR, M.A. GIVE'ON, V. *ALGEBRA AUTOMATA II: THE CAT-000019
 ARBIR, M.A. GIVE'ON, V.*CONTRIBUTIONS TO THE THEORY OF 000011
 ARBIR, L.A. KAHN, R.M.*A DEVELOPMENTAL MODEL OF INFOR 000015
 ARBIR, M.A. KAHN, R.M.*A CYBERNETIC APPROACH TO MENTAL 000016
 ARNOLD, L.*ON THE ASYMPTOTIC DISTRIBUTION OF THE EIGEN 000014
 ATKINS, D.E.*THE THEORY OF IMPLEMENTATION OF FRT DIVIS 000013
 AUTOMATA AND MONADIC-ALGEBRAS.*ILLINOIS U. COMP.SCI. 000022
 AUTOMATA II: THE CATEGORICAL FRAMEWORK FOR DYNAMIC A 000019
 AUTOMATA.*BRIEFING SC L.PES.LARS.*MATH. NOTE 497.*1967. 000020
 AUTOMATA.*ST. U.E.E.DEPT.*1967. 000011
 AUTOMATON AND ITS QUOTIENT AUTOMATA.*ILLINOIS U. COMP. 000021
 BAECCKER, R.M.*PLANAR-REPRESENTATIONS OF COMPLEX GRAP 000010
 BALUEV, A.N.*THREE PAPERS ON CHAPLYGIN-METHODS AND NO 000012
 BALZER, R.M.*STUDIES CONCERNING MINIMAL TIME SOLUTIONS 000009
 BAYER, Q.*AUTOMORPHISM GROUPS AND QUOTIENTS OF STRONGL 000022
 BAYER, R.*ENDOMORPHISMS AND CONGRUENCES OF AUTOMATA 000020
 BAYER, R.*THE AUTOMORPHISM GROUP OF A STRONGLY CONNECT 000021
 BOEING SC L.RES.LABS.*MATH. NOTE 457.*1967. 000020
 CARNESIE INST. OF TECHNOLOGY.*1966. 000009
 CHAPLYGIN-METHODS AND NONLINEAR-EQUATIONS. *YARYLAN 000012
 CHILD.*ST. U.E.E.DEPT.*1967. 000015
 COROL SPECIFICATIONS.*AMERICAN STANDARDS ASSN.*196 000002
 CYBERNETIC APPROACH TO MENTAL DEVELOPMENT. ST. U. E.E. 000016
 CYBERNETICS IN EUROPE AND THE U.S.S.R.*P.E.C. RESEAR 000017
 DIVISION.*ILLINOIS. COMP.SCI.DEPT.*RPT. 230.*1967. 000013
 EIGENVALUES OF RANDOM MATRICES: WISCONSIN U. MATH.RE 000014
 FIRING-SQUAD SYNCHRONIZATION PROBLEM: CARNESIE INST. 000009
 GIVE'ON, Y. *ALGEBRA AUTOMATA II: THE CATEGORICAL FR 000019
 GIVE'ON, Y.*CONTRIBUTIONS TO THE THEORY OF ALGEBRAS C 000011
 GRAPHS.*M.I.T. LINCOLN LAB.*T.N.1967-1967. 000010
 HYPERPOLIC-EQUATIONS WITH DISCONTINUOUS INITIAL-VALUES. 000004
 ILLINOIS U. COMP.SCI.DEPT.*RPT. 204.*1966. 000022
 ILLINOIS U. COMP.SCI.DEPT.*RPT.196.*1966. 000021
 ILLINOIS U. COMP.SCI.DEPT.*RPT. 255.*1968. 000013
 ILLINOIS. COMP.SCI.DEPT.*RPT.*230.*1967. 000013
 INFORMATION-PROCESSING IN THE CHILD.*ST. U.E.E.DEPT.* 000015

Figure 4. Sample Ito Category Ito Access Output (Sheet 3 of 4)

KEYWORD LISTING

000004 REFERENCE SCHEMES FOR HYPERBOLIC-EQUATIONS WITH DISCONTINUOUS INITIAL-VALUES.* UPPSALA U. COMP.SCI.DEPT.*RPT.5.*1967. 000004
000015 KAHN, R.M.*A DEVELOPMENTAL MODEL OF INFORMATION-PROCESS 000015
000016 KAHN, R.M.*CYBERNETIC APPROACH TO MENTAL DEVELOPMENT 000016
000017 LISP-2 LANGUAGE SPECIFICATIONS.* SYSTEM DEVELOPMENT CO 3 0 0 0 3
000018 M.I.T. LINCOLN LAB.*N. 1967-1*1967. 000018
000019 MARYLAND U. COMP.SCI. CENTER.* TR-64-R.*1964. 000019
000020 MARYLAND U. COMP.SCI. CENTER.*TR-67-41.*1967. 000020
000021 MATRICES.* WISCONSIN U. MATH.RES. CENTER.*MRC TECH. SU 000021
000022 MONASTIC-ALGEBRAS.* ILLINOIS U. COMP.SCI.DEPT.*RPT.2 0 000022
000023 MOORE, F.H.*AN EXTENSION OF THE NEWTON-KANTOROVICH ME 000023
000024 NEUROCYBERNETICS.* ST. U. E.E.DEPT.* 1967. 000024
000025 NEWTON-KANTOROVICH METHOD FOR SOLVING NONLINEAR-EQUAT 000025
000026 NONLINEAR-EQUATIONS.* MARYLAND U. COMP.SCI. CENTER.* 000026
000027 P.E.C. RESEARCH ASSOCIATES, INC.*AFOSR-65-1412.*1965. 000027
000028 PATTERN-RECOGNITION: I-COMPUTER PROGRAM FOR GENERATING 000028
000029 PULSE SEQUENCES.* ILLINOIS COMP.SCI.DEPT.*RPT.255.*196 000029
000030 ROSENFELD, A. SIMENSON, N.F.* PATTERN-RECOGNITION: I. CO 000030
000031 SELF-REPRODUCING-AUTOMATA: SOME IMPLICATIONS FOR THEOR 000031
000032 SIMENSON, N.F.* PATTERN-RECOGNITION: I. COMPUTER PROG 000032
000033 SNODGRASS, A.P.*APPLIED LOGIC CORP.* 1967. B - W 000033
000034 ST. U. E.E.DEPT.* 1967. 000034
000035 ST. U. E.E.DEPT.* 1967. 000035
000036 ST. U. E.E.DEPT.* 1967. 000036
000037 ST. U. E.E.DEPT.* 1967. 000037
000038 ST. U. E.E.DEPT.* 1967. 000038
000039 ST. U. E.E.DEPT.* 1967. 000039
000040 SYNTHETIC-PATTERNS.* MARYLAND U. COMP.SCI. CENTER.* 000040
000041 SYSTEM DEVELOPMENT CORP.*TR-3417/200/00.*1967. 000041
000042 UPPSALA U. COMP.SCI.DEPT.*RPT.5.*1967. 000042
000043 WISCONSIN U. MATH.RES. CENTER.*MRC TECH. SUMMARY PP 000043
000044 WISCONSIN U. MATH.RES. CENTER.*MRC TECH. SUMMARY PP. 000044

Figure 4. Sample NoCategory Mode ACCESS Output (Sheet 4 of 4)

```

1.
2.
3.
4.
5.
6.
7.
8.
9.
10.
11.
12.
13.
14.
15.
16.
17.
18.
19.
20.
21.
22.
23.
24.
25.
26.
27.
28.
29.
30.
31.
32.
33.
34.
35.
36.
37.
38.
39.
40.
41.
42.
43.
44.
45.
46.
47.
48.
49.
50.
51.
52.
53.
54.
55.
56.
57.
58.
59.

```

*LIE
 *CATEGORY NAMES
 C C C M P I L E S
 D C A T A STRUCTURES
 *END CATEGORY NAMES
 *ACCESSION ENTRIES
 R0162 SWISTH, A. AND H. JEWBER, "EULER: A GENERALIZATION OF RALGOL AND ITS CD
 + FORMAL DEFINITION," GADM, JAN. 1966, PG. 13. PRECEDENCE, PARSING
 *END ACCESSION ENTRIES

Figure 5. Sample ACCESS Input To Show Sort Keys

S C R I P T	KEY	P R I N T	L I N E
BC+A	... (NOT USED)	...	***** CATEGORY -- C ... C L M P I L E Q S
BC+R	... (NOT USED)	...	
BC+C	... (NOT USED)	...	
BD+A	... (NOT USED)	...	***** CATEGORY -- D ... D A T A S T R U C T U R E S
BD+E	... (NOT USED)	...	
BD+C	...	...	(ACCESS TITLE)
ARC182 1	...	...	(1ST CATEGORY)
BCW1R1P, A. AND H. JEWBER	...	...	(2ND CATEGORY)
BCW1R1P, A. AND H. JEWBER	...	...	(CONTINUATION IMAGE FOR ACCESSION LISTING)
ARC182 2	...	...	(1ST KEYWORD IN ACCESSION ENTRY)
BCW1R1P, A. AND H. JEWBER	...	...	(2ND KEYWORD IN ACCESSION ENTRY)
CAREER, "EULER: A GENF	...	...	(3RD KEYWORD IN ACCESSION ENTRY)
CALCCLANDIT S FORMAL D	...	...	(4TH KEYWORD IN ACCESSION ENTRY)
CPRECEDENCE, PARSING	...	...	(5TH KEYWORD IN ACCESSION ENTRY)
CPARSING	...	...	

Figure 5. Sort Keys Before Sort

